The Journal of CIEES
Volume 2, Issue 2, December 2022, pp. 7 – 12, ISSN: 2738-7283 (Print), 2738-7291 (Online)
DOI: 10.48149/jciees.2022.2.2.1

Spatial Machine Learning Personal Mobility Predictive Model Trained with Smartphone-Collected Trajectory Data

Boyan Stoyanovski [1*], Teodor B. Iliev [1], Radovan Cesarec [2] and Renato Filjar [3]
[1] University of Ruse/Department of Telecommunications, Ruse, Bulgaria
[2] Krapina University of Applied Sciences, Krapina, Croatia
[3] Krapina University of Applied Sciences/Laboratory for Spatial Intelligence, Krapina, Croatia

*E-mail of corresponding author: stoyanovskiboyan@gmail.com

Abstract: Individual and group mobility is an essential information for numerous segments of technology (including transport and logistics), society, and economy. The ability of telecommunications devices, such as smartphones, to collect accurate and reliable data on personal mobility with the embedded sensors, inspires research in personal mobility. We confirm the ability of suitably defined indicators to compare sets of trajectories, and identify outliers/differences among the individual ones. Furthermore, we demonstrate development of a machine learning (ML) regression predictive model based on experimental data collected on the real urban environment of the city of Krapina, Croatia, suitable for utilisation in personal mobility analysis, and traffic and transport planning and optimisation.

Keywords: GNSS, trajectory, kinematics, machine learning, predictive model, mobility.

Received 11 October 2022
Accepted 10 November 2022
Published 19 December 2022

1. Introduction

Individual and group mobility is an essential information for numerous segments of technology (including transport and logistics), society, and economy. Different methods for mobility analysis and estimation exist, based on a wide range methods for different data collection. The ability of telecommunications devices, such as smartphones, to collect accurate and reliable data on personal mobility with the embedded sensors, inspires research in personal mobility. A Rich set of mobility related sensors, including a GNSS receiver and accelerometers, embedded in modern smartphones render a fundamental framework in support of the telecommunications-related Location-Based Services, but may be exploited in a range of other applications.

The fact leads to attempt to analyse data in a systematic manner, using methods of trajectory analysis [1] deployed as a bespoke software developed for the purpose of this research using the special library trajr [2] in the R environment for statistical computing [3] using its RStudio Graphical User Interface (GUI) [4]. We attempted to confirm the hypothesis of ability of suitably defined indicators to compare sets of trajectories, and identify outliers/differences among the individual ones. Furthermore, we aimed at showing that the analysis of multiple trajectories on similar paths in a given transport/traffic network may yield the objective indices of variance in trajectories, thus justifying consideration of their utilisation for traffic and transport planning and optimisation. In practical demonstration of the assumption, the trajectory analysis presented allows us to develop a machine learning (ML) regression predictive model based on experimental data collected on the real urban environment of the city of Krapina, Croatia.

The research presented comprises tasks as follows: Global Navigation Satellite System (GNSS) position estimate collection in scenario of individual mobility, data cleaning and collation, aggregation of GNSS position estimations into trajectories, spatial data analysis and mobility analysis, graphical presentation and interpretation of the results. Methodology, research results, and discussion, resulting from this research, are presented in the rest of this document.

2. Material and research scenario

GNSS position estimates, taken by a GNSS receiver embedded in an Android smartphone, served as a raw material. In the period between 6 July, 2022 to 20 August, 2022, with 119 records in total, a path was walked by one of us (BS) between the place in Dolac Street, Krapina, Croatia and the Krapina University of Applied Sciences (KUAS) building, through the city centre of Krapina, Croatia, as shown in Figure 1. Figure 1 was prepared using a tailored programme developed for the purpose of this research in the open-source R environment for statistical computing [5], and utilised spatial data from the open-access OpenStreetMap database, and the R-library called leaflet. Slight diversions and variations of the path walked were introduced intentionally to allow for variances in the trajectories.

The AndroSensor, an Android-based application, was used to collect GNSS position estimates twice every second (sampling period of 0.5 s).

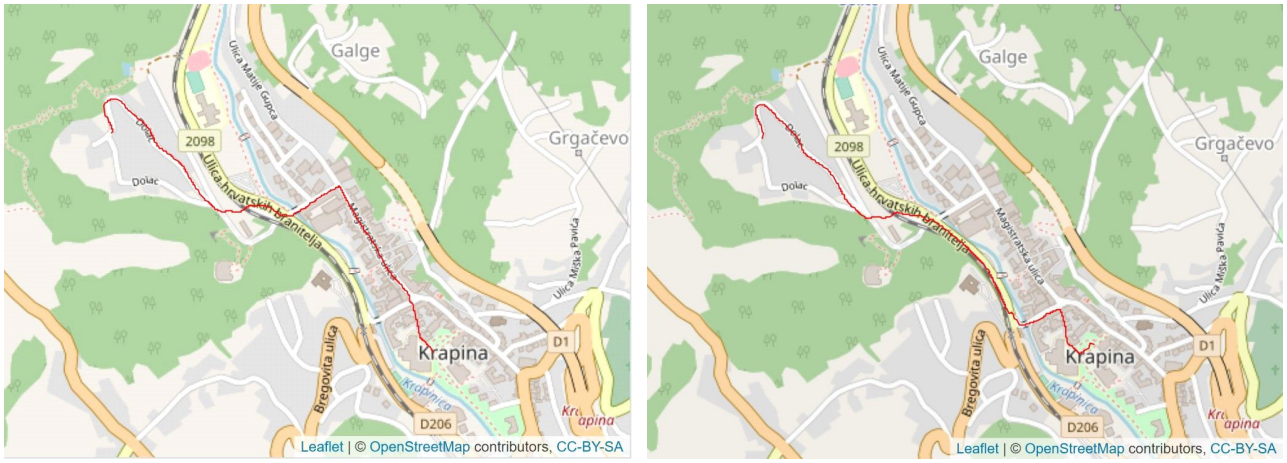

Fig. 1. Two individual paths commonly taken during research, from the starting point at Dolac Street, Krapina to the end-point of KUAS facility in the Krapina city centre. The red line represents a trajectory assembled from a set of GNSS position estimates as a spatial trajectory object, in the R environment for statistical computing.

A set of GNSS position estimates collected during every round of walk, was used to define a trajectory. AndroSensor reads and stores readings of other smartphone sensors, such as accelerometers and gyros, but those have not been used in this research, but remain in the experimental dataset collected to be exploited in future ones.

Collected data are examined for quality, samples with rare missing observations (NA) are removed, and data collated into a machine-readable format. Collected, aggregated, and collated data sets are stored in a CSV database.

3. Methodology for trajectory analysis and modelling

Formal description of trajectory analysis was performed based on the statistical methodology outlined in related references [1] and [2]. The following indices describing individual trajectories were used in this research as trajectory indices (features, statistical variables describing a trajectory): (1) (diffusion) distance of points (position estimates) on individual trajectory; (2) trajectory straightness; (3) trajectory fractal dimension; (4) mean travel velocity, and (5) duration of the path travelled for an individual trajectory.

Indices are defined as follows.

(Diffusion) distance is defined in terms of connectivity, or similarity between two trajectory points x_i and x_j, observed with the time separation of t, and belonging to the data set X as a connected graph [5]. A measure space (X, A, μ) is to be defined on X, with A denoting a set of sub-sets of X, and μ denoting a measure on (X, A). Let us express the connectivity k between two arbitrary data points of X, x and y, in terms of definition of probability $p(x, y)$ of travelling from x to y in a single random walk step. The probability may be expressed using a suitable kernel function, a Gaussian kernel being a function used in this research, as given in (1), with ε as a parameter.

$$k(x,y)=\exp\left[-\frac{x-y}{\varepsilon}\right] \tag{1}$$

With (X, k) defined as above, a reversible discrete time Markov chain may be developed on X, as given in (2).

$$d(x)=\int_X k(x,y)d\mu(y) \tag{2}$$

Finally, the probability of a single-step random walk may be determined as given in (3).

$$p(x,y)=\frac{k(x,y)}{d(x)} \tag{3}$$

The formal expression of diffusion distance may be then defined with (4) and (5).

$$D_t(x_i,x_j)^2=\sum_y \frac{\left(\left(p(y,t|x_i)\right)-\left(p(y,t|xj)\right)\right)^2}{\phi_0(y)} \tag{4}$$

$$\phi_0(y)=d\frac{(y)}{\sum_{z\in X} d(z)} \tag{5}$$

A (diffusion) distance index value is determined in this research based on the method (4) & (5) deployment in the *trajr* R library [6], and assigned to every trajectory of the 119 element set.

Straightness of a trajectory is considered as a measure of efficiency of a direct walk [2], [6], and defines as a ratio between the shortest (Euclidean) distance between the start and the end points, D, and the actual length of a trajectory, L, calculated as a summation of distances of neighbouring position samples on a trajectory. Straightness index is determined as given in (6) [2], [6].

$$S=\frac{D}{L} \tag{6}$$

A straightness index value is determined in this research based on the method (6) deployment in the *trajr*

R library, and assigned to every trajectory of the 119 element set.

Fractal dimension of a trajectory is regarded as a measure of a geometrical smoothness of the trajectory concerned. It allows a researcher to get a better insight in effects of local topography on trajectory development, without the need for a formal description of the environment surrounding a walker. Estimation of the fractal dimension index is based on the box-counting method proposed by B Mandelbrot in [7], and further refined for ecological/trajectory research by in [8]. In this research, a fractal dimension index value is determined based on the method [8] deployed in the *trajr* R library [6], and assigned to every trajectory of the 119 element set.

Mean travel velocity extends an average value of velocity along the trajectory concerned. The *trajr* R library determines the mean travel velocity for a given trajectory as a vector [6]. This research utilises the length (absolute value) of this vector as a descriptive trajectory index.

Duration of the path travelled for an individual trajectory is measured directly for every trajectory as the time difference between timestamps of end point (last position estimate in trajectory) and starting point (first position estimate in trajectory), respectively.

Trajectory is described by its points, and the values of the above-stated indices. It should be noted that the information on the trajectory context (environment) is embedded indirectly within the set of indices. Multiple trajectory analysis is performed using the trajectory indices that were analysed using exploratory statistical methods applied on sets of trajectory feature values, as statistical variables [9].

This research suggests that the information content of trajectory indices allows for inference in relation to trajectories. This is justified with the demonstration of the application of machine learning methods for prediction of duration of the path travelled (one of the indices, serving as a target statistical variable) based on the knowledge of values of the other indices (those becoming predictors) for the specified trajectory. In the sense of supervised machine learning, several methods for regression predictive model development are applied on the Krapina experimental data set, with the aim of development of the predictive model that returns duration of the path travelled based on known values of diffusion distance, straightness, fractal dimension, and mean travel velocity, which characterise the manner the path was walked. The machine learning model development methods utilised in this research are, as follows: (1) k-Nearest Neighbours (KNN), (2) Support Vector Machine (SVM), (3) Classification And Regression decision Tree (CART), and (4) Random Forest (RF). Detailed outline of machine learning methods utilised may be found elsewhere in excellent references, including [10], and [11]. Particularly, Ceja in [11] outlined the foundations for machine learning in trajectory analysis and prediction, enhancing the motivation for the research presented. The four machine learning methods utilised will be described here comprehensively.

The k-Nearest Neighbours KNN) algorithm is a simple non-parametric machine learning method that develops regression prediction model based on classification of samples using a reference distance metrics, and assignment of an average value of the target variable in the class to which the predicted sample belongs.

The Support Vector Machine (SVM) method works in a similar manner to the KNN method, as it first classifies samples into classes, and determine the average target variable value of the class to which the predicted sample belongs. However, the means of classification is different from the KNN method, as SVM clusters data using a distance metrics, and then determines the separation plane between the groups. Using the kernel approach (1), the separation plane may be applied to non-linear cases.

The CART method is essentially a decision tree method. It first analyses 'vagueness' in values of variables, and determines the order of appearance of variables along the structure of decision tree. The CART model returns one of the selected outcomes (target variable values) determined during the model development process based on experimental observations.

Finally, the Random Forest (RF) method develops the predictive model by training a large number of decision trees, say 1000, every single one using a different sub-set of original data (observations of statistical variables). Faced with the new case (sample), decision trees will respond with their decisions, and their average is returned as the outcome of the Random Forest model. Compared with the other methods, the RF returns more robust model, which is very accurate, and not prone to over-fitting (modelling noise, rather than signal).

This research utilises deploymet of the above-stated predictive model development methods in the R environment for statistical computing, and in particular the specific cart R library [12].

Machine learning methods return models in an automated manner. Naturally, the process requires a methodical assessment of performances of the developed models, and selection of one that predicts in the optimal manner [13]. Model performance assessment is performed using a separate data set, not used in the model development procedure. This research splits the original set of statistical variables values into a *training set*, comprising randomly selected samples of amounting to 80% of the *original set*, and a testing set, comprising the rest. The 80-20 split was originally proposed by Pareto [10].

The *caret* R package returns not only the model of the desired structure, but a set of performance indicators that allows of the model performance assessment and comparison with the other models. A good model generally balances its ability to model bias and variance in the phenomenon described by observations. In that terms, this research applies three essential criteria (performance indicators) to select the optimal prediction model for the purpose of prediction of duration of trajectory based on (diffusion) distance of points (position estimates) on individual trajectory, trajectory straightness, trajectory fractal dimension, and mean travel velocity. Performance indicators are derived from the so-called

residual analysis, during which the differences between predicted target values obtained from the model $x^{(p)}_i$, and actually observed target values for the same predictor values xi, are determined as prediction errors (residuals). Two performance indices describe the success in description of bias: the Mean Absolute Error (MAE), and the Root Mean Square Error (RMSE).

The MAE is determined using (7) on the *N* samples in the testing sub-set.

$$MAE = \frac{\sum_i \left| x_i^{(p)} - x_i \right|}{N} \tag{7}$$

In the manner similar to MAE, the RMSE is defined using (8).

$$RMSE = \sqrt{\frac{\sum_i^N \left(x_i^{(p)} - x_i \right)^2}{N}} \tag{8}$$

An accurate model extends small values of both MAE and RMSE.

The *adjusted coefficient of determination* (*adjR2*) is a widely accepted performance index that describes the model's ability to follow the original variance in observations. The *adjR2* index is derived from its parent *coefficient of determination R2*, in the manner, as follows. The R2 index is defined by (9), with variables denoted as in (7), and with $\bar{x}$ denoting the arithmetic mean of x_i.

$$R2 = \frac{\sum_i^N \left(x_i^{(p)} - \bar{x} \right)^2 - \sum_i^N \left(x_i^{(p)} - x_i \right)^2}{\sum_i^N \left(x_i^{(p)} - \bar{x} \right)^2} \tag{9}$$

The R2 index is dependent on the number of samples used for the assessment and the number of predictors in the model. The shortcoming prevents the comparison of models with different number of predictors. Comparison is viable using the *adjR2* index, defined as in (10), with R^2 determined by (9), n denoting number of samples in dataset, and *k* denoting the number of independent predictors of the model.

$$adjR2 = 1 - \frac{\left(1 - R^2\right)(n-1)}{(n-k-1)} \tag{10}$$

Adjusted coefficient of determination may be interpreted as the proportion of the original variance in phenomenon/observations described by the model (for instance, adjR2=0.92 means 92% of the original variance is described by the model).

4. Research results

This research exploits observations described in Section 2, using the methodology and the bespoke R software tools described in Section 3.

First, the comprehensive results of statistical analysis of collected data are presented, with utilisation of five statistical variables presenting trajectory indices: (1) (diffusion) distance of points (position estimates) on individual trajectory; (2) trajectory straightness; (3) trajectory fractal dimension; (4) mean travel velocity, and (5) duration of the path travelled for an individual trajectory, with their scalar values assigned for every of 119 trajectories under consideration.

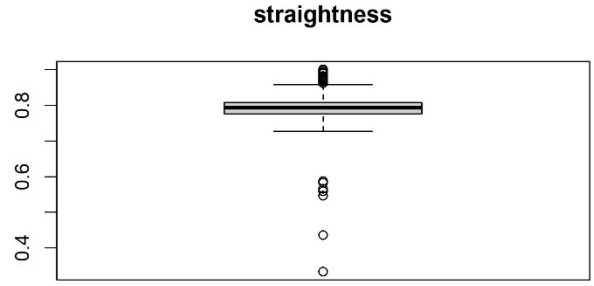

Fig. 2. Box-plot of straightness index values for 119 trajectories.

Figure 2 depicts the box-plot of straightness index values for 119 trajectories.

The box-plot reveals a close similarity in straightness for majority of trajectories, as well as several well-defined outliers. Particular outliers may be easily identified by a slight expansion of the R script.

Figure 3 depicts the box-plot of (diffusion) distance index for 119 trajectories.

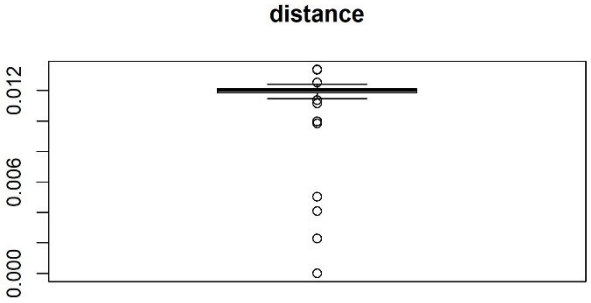

Fig. 3. Box-plot of distance index for 119 trajectories.

It shows even more emphasised unanimity of a majority of trajectories, and even more pronounced outliers.

Figure 4 depicts the box-plot of duration index for 119 trajectories

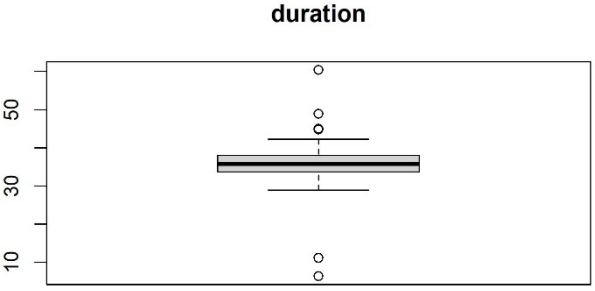

Fig. 4. Box-plot of duration index for 119 trajectories.

Times needed for performing the trajectory pass differ. Still, the majority takes very similar time, while the outliers, comprising significantly different paths, or additions to the common one, are easily identified.

Figure 5 depicts the box-plot of fractal dimension index of 119 trajectories.

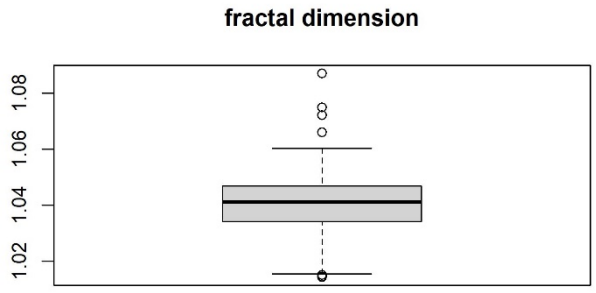

Fig. 5. Box-plot of fractal dimension index of 119 trajectories.

Fractal dimension extends the level of broken lines consisting the trajectory under observation. Although the box-plot extends several outliers, the set examined seems to extend smoothness.

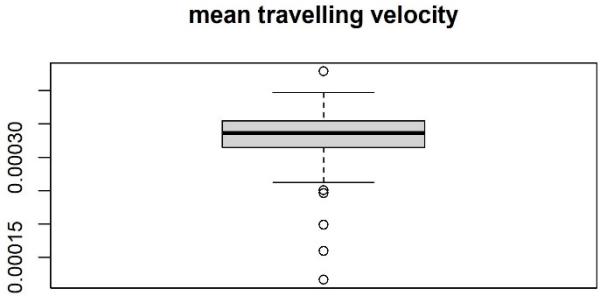

Fig. 6. Box-plot of mean travelling velocity index, across the trajectory concerned, of 119 trajectories.

Mean travelling velocity extends similarity, while still several outliers exist.

The process of regression model development requires the examination of linear association between statistical variables. The correlogram of the variables involved is presented in Figure 7.

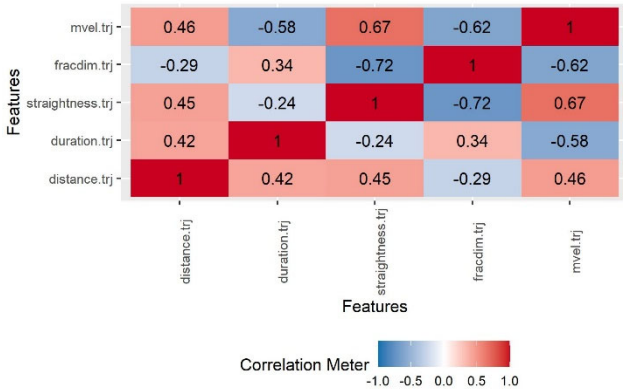

Fig. 7. Correlogram of statistical variables (features: predictors and target) concerned.

The correlogram depicts a general lack of correlation (linear association) both between predictors (in support of linear regression model development), and target and individual predictors (an obstacle for linear regression model development).

The relation between target (duration) and individual predictors is more visible in the segmented boxplots panel, depicted in Figure 8.

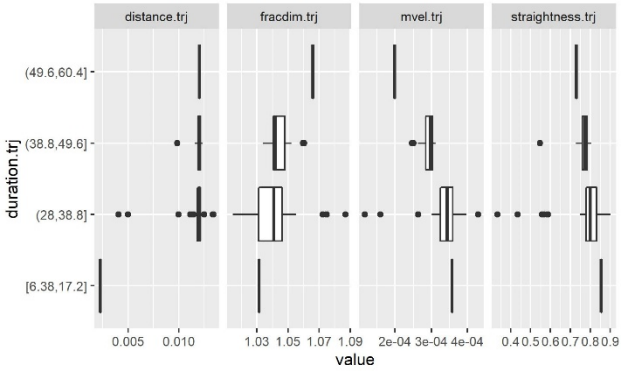

Fig. 8. Panel of box-plot diagrams with segmentation rendered in relation to target (predicted) variable *duration.trj*.

Although diffusion distance does not extend a considerable alignment of box-plots with the ranges of target values, alignment is visible with the other statistical variables assumed as predictors.

The four machine learning predictive model methods: k-Nearest Neighbours (KNN), Support Vector Machine (SVM), Classification And Regression decision Tree (CART), and Random Forest (RF), are applied to yield four predictive models. Performance of the four models are assessed by three criteria: MAE, RMSE, and adjR2, as outlined in Section 3. The performance assessment results are summarised graphically in Figure 9.

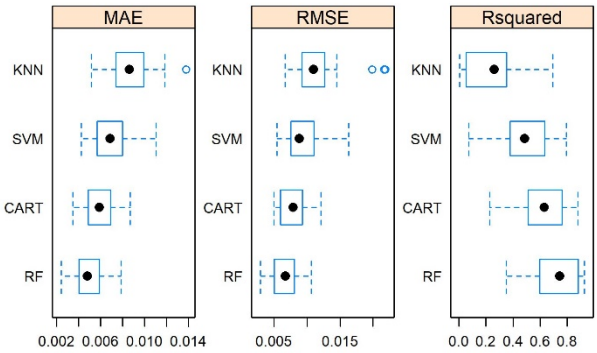

Fig. 9. Performance indices Mean Absolute Error (MAE), Root-Mean Square Error (RMSE), and Rsquared for candidate predictive models developed using k-Nearest Neighbours (KNN), Support Vector Machine (SVM), Classification And Regression decision Tree (CART), and Random Forest (RF) machine learning methods.

Model performance assessment is conducted using the cross-validation approach [11], which yields a set of performance indices values related to different testing sub-sets used. Figure 9 depicts box-plots of such sets. The performance analysis results returns the Random Forest model as the optimal predictive model for the problem of trajectory duration prediction, given the predictor variables values. The RF model evidently extends the smallest median MAE and RMSE, while at the same time covering the largest amount of variance, with median of approx. 75%.

5. Discussion and contribution.

This research aimed at assessment of the ability of several trajectory indices to serve as machine learning descriptors of mobility in the sense of trajectory

classification. We have shown the scalar geometry-based indices describe in trajectory accurately and robustly, while allowing for prediction process without the need for a local topography. The finding renders them possible predictor candidates for machine learning trajectory classification process.

This research assumed the GNSS-based position estimation as flawless, thus presuming the variance in trajectory indices (descriptors) being caused by small intentional variations in path travelled. However, variances in trajectories may result in significant errors in GNSS position estimation caused by non-accounted for effects of space weather/ionospheric disturbances/storms, multipath, or deliberate cyberattacks (spoofing or jamming).

This research will continue with assessment of GNSS position estimation error effects on trajectory identification and classification, and the risk assessment of potential misclassifications.

Statement of open access to r software and data

Authors have made openly available the collection of observations and the developed R-based software to interested scientists and researchers. Citation & access (after 1st December, 2022):

Stoyanovski, B., Iliev, T., Cesarec, R., Filjar, R. (2022). Supplementary material (data & R script) for the manuscript authored by B Stoyanovski, T Iliev, R Cesarec, R Filjar. figshare. Dataset. doi: https://doi.org/10.6084/m9.figshare.21205304.v2

Statement of contribution

BS and RF conceived and designed study. BS and RC designed scenario and observation collection methodology. BS collected observations. BS, RF, and TI designed methodology for trajectory analysis and predictive modelling. BS, RF, and TI developed the R-based software. BS and TI analysed trajectores, developed and assessed predictive model. All authors discussed results and drew the conclusion.

References

[1] Zheng, Y. (2015). Trajectory Data Mining: An Overview. *ACM Transactions on Intelligent Systems and Technology*, 61(3), 1–41. doi: 10.1145/2743025.

[2] McLean, D J, and Skowron Volponi, M A. (2018). trajr: An R package for characterisation of animal trajectories. *Ethology*, 124, 440–448. doi: 10.1111/eth.12739.

[3] R project team. (2022). *R version 4.2.1*. Available at: https://cloud.r-project.org/.

[4] RStudio. (2022). *RStudio version 2022.07.0 Build 548* Available at: https://www.rstudio.com/products/RStudio/#Desktop.

[5] Nadler, B, Lafon, S, Coifman, R R, and Kevrekidis, I G. (2005). *Diffusion Maps, Spectral Clustering and Eigenfunctions of Fokker-Planck operators*. arXiv.org. doi: 10.48550/arXiv.math/0506090.

[6] McLean, D J. (2020). *Animal trajectory analysis with trajr - trajr vignette*. Available at: https://cran.r-project.org/web/packages/trajr/vignettes/trajr-vignette.html.

[7] Mandelbrot, M. (1967). How Long Is the Coast of Britain? Statistical Self-Similarity and Fractional Dimension. Science, 156(3775), 636-638. doi: 10.1126/science.156.3775.636.

[8] Sugihara, G, and May, R M. (1990). Applications of fractals in ecology. *Trends in Ecology and Evolution*, 5(3), 79-86. doi: 10.1016/0169-5347(90)90235-6.

[9] Hartvigsen, G. (2021). *A Primer in Biological Data Analysis and Visualization Using R* (2nd ed). Columbia University Press. New York, NY. ISBN 9780231554404.

[10] Murphy, K P. (2022). *Probabilistic Machine Learning: An Introduction* (2nd ed). MIT Press. Cambridge, MA. ISBN 978026236930.

[11] Ceja, E G. (2021). *Behavior Analysis with Machine Learning and R: A Sensors and Data Driven Approach*. CRC Press. Boca Raton, FL. ISBN 978-1032067049. Available at: https://enriquegit.github.io/behavior-free/.

[12] Kuhn, M. (2019). The caret package. Available at: https://topepo.github.io/caret/.

[13] Biecek, P, and Burzykowski, T. (2021). *Explanatory Model Analysis: Explore, Explain, and Examine Predictive Models, With examples in R and Python*. CRC Press. Boca Raton, FL. ISBN 9780367135591.